Attention Optimization for Abstractive Document Summarization

Min Gui[1], Junfeng Tian[1], Rui Wang[1], Zhenglu Yang[2]
[1] Alibaba Group, China
[2] School of Computer Science, Nankai University, China
{guimin.gm, tjf141457, masi.wr}@alibaba-inc.com
yangzl@nankai.edu.cn

Abstract

Attention plays a key role in the improvement of sequence-to-sequence-based document summarization models. To obtain a powerful attention helping with reproducing the most salient information and avoiding repetitions, we augment the vanilla attention model from both local and global aspects. We propose an attention refinement unit paired with local variance loss to impose supervision on the attention model at each decoding step, and a global variance loss to optimize the attention distributions of all decoding steps from the global perspective. The performances on the CNN/Daily Mail dataset verify the effectiveness of our methods.

1 Introduction

Abstractive document summarization (Rush et al., 2015; Nallapati et al., 2016; Tan et al., 2017; Chen and Bansal, 2018; Celikyilmaz et al., 2018) attempts to produce a condensed representation of the most salient information of the document, aspects of which may not appear as parts of the original input text. One popular framework used in abstractive summarization is the sequence-to-sequence model introduced by Sutskever et al. (2014). The *attention* mechanism (Bahdanau et al., 2014) is proposed to enhance the sequence-to-sequence model by allowing salient features to dynamically come to the forefront as needed to make up for the incapability of memorizing the long input source.

However, when it comes to longer documents, basic attention mechanism may lead to distraction and fail to attend to the relatively salient parts. Therefore, some works focus on designing various attentions to tackle this issue (Tan et al., 2017; Gehrmann et al., 2018). We follow this line of research and propose an effective attention refinement unit (ARU). Consider the following case. Even with a preliminary idea of which parts of source document should be focused on (attention), sometimes people may still have trouble in deciding which exact part should be emphasized for the next word (the output of the decoder). To make a more correct decision on what to write next, people always adjust the concentrated content by reconsidering the current state of what has been summarized already. Thus, ARU is designed as an update unit based on current decoding state, aiming to retain the attention on salient parts but weaken the attention on irrelevant parts of input.

The de facto standard attention mechanism is a soft attention that assigns attention weights to all input encoder states, while according to previous work (Xu et al., 2015; Shankar et al., 2018), a well-trained hard attention on exact one input state is conducive to more accurate results compared to the soft attention. To maintain good performance of hard attention as well as the advantage of end-to-end trainability of soft attention, we introduce a local variance loss to encourage the model to put most of the attention on just a few parts of input states at each decoding step. Additionally, we propose a global variance loss to directly optimize the attention from the global perspective by preventing assigning high weights to the same locations multiple times. The global variance loss is somewhat similar with the coverage mechanism (Tu et al., 2016; See et al., 2017), which is also designed for solving the repetition problem. The coverage mechanism introduces a coverage vector to keep track of previous decisions at each decoding step and adds it into the attention calculation. However, when the high attention on certain position is wrongly assigned during previous timesteps, the coverage mechanism hinders the correct assignment of attention in later steps.

We conduct our experiments on the CNN/Daily Mail dataset and achieve comparable results on ROUGE (Lin, 2004) and METEOR (Denkowski and Lavie, 2014) with the state-of-the-art models.

Our model surpasses the strong pointer-generator baseline (w/o coverage) (See et al., 2017) on all ROUGE metrics by a large margin. As far as we know, we are the first to introduce explicit loss functions to optimize the attention. More importantly, the idea behind our model is simple but effective. Our proposal could be applied to improve other attention-based models, which we leave these explorations for the future work.

2 Proposed model

2.1 Model Architecture

We adopt the Pointer-Generator Network (PGN) (See et al., 2017) as our baseline model, which augments the standard attention-based seq2seq model with a hybrid pointer network (Vinyals et al., 2015). An input document is firstly fed into a Bi-LSTM encoder, then an uni-directional LSTM is used as the decoder to generate the summary word by word. At each decoding step, the attention distribution a_t and the context vector c_t are calculated as follows:

$$e_{ti} = v^T \tanh(W_h h_i + W_s s_t + b_{attn}) \quad (1)$$

$$a_t = \text{softmax}(e_t) \quad (2)$$

$$c_t = \sum_{i=1} a_{ti} h_i \quad (3)$$

where h_i and s_t are the hidden states of the encoder and decoder, respectively. Then, the token-generation softmax layer reads the context vector c_t and current hidden state s_t as inputs to compute the vocabulary distribution. To handle OOVs, we inherit the pointer mechanism to copy rare or unseen words from the input document (refer to See et al. (2017) for more details).

To augment the vanilla attention model, we propose the Attention Refinement Unit (ARU) module to retain the attention on the salient parts while weakening the attention on the irrelevant parts of input. As illustrated in Figure 1, the attention weight distribution a_t at timestep t (the first red histogram) is fed through the ARU module. In the ARU module, current decoding state s_t and attention distribution a_t are combined to calculate a refinement gate r_t:

$$r_t = \sigma(W_s^r s_t + W_a^r a_t + b_r) \quad (4)$$

where σ is the sigmoid activation function, W_s^r, W_a^r and b_r are learnable parameters. r_t represents how much degree of the current attention should
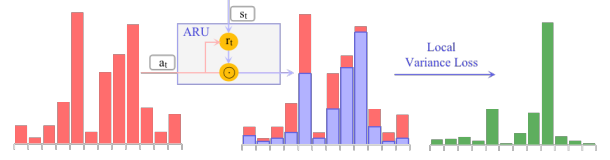

Figure 1: The process of attention optimization (better view in color). The original attention distribution (red bar) is updated by the refinement gate r_t and attention on some irrelevant parts are lowered. Then the updated attention distribution (purple bar) is further supervised by a local variance loss and get a final distribution (green bar).

be updated. Small value of r_{ti} indicates that the content of i-th position is not much relevant to current decoding state s_t, and the attention on i-th position should be weakened to avoid confusing the model. The attention distribution is updated as follows (the symbol $\odot$ means element-wise product):

$$a_t^r = r_t \odot a_t \quad (5)$$

2.2 Local Variance Loss

As discussed in section 1, the attention model putting most of attention weight on just a few parts of the input tends to achieve good performance. Mathematically, when only a small number of values are large, the shape of the distribution is sharp and the variance of the attention distribution is large. Drawing on the concept of variance in mathematics, local variance loss is defined as the reciprocal of its variance expecting the attention model to be able to focus on more salient parts. The standard variance calculation is based on the mean of the distribution. However, as previous work (Huang et al., 1979; Jung et al., 2018) mentioned that the median value is more robust to outliers than the mean value, we use the median value to calculate the variance of the attention distribution. Thus, local variance loss can be calculated as:

$$var(a_t^r) = \frac{1}{|D|} \sum_{i=1}^{|D|} (a_{ti}^r - \hat{a_t^r})^2 \quad (6)$$

$$\mathcal{L}_L = \frac{1}{T} \sum_t^T \frac{1}{var(a_t^r) + \epsilon} \quad (7)$$

where $\hat{\cdot}$ is a median operator and ϵ is utilized to avoid zero in the denominator.

2.3 Global Variance Loss

To avoid the model attending to the same parts of the input states repeatedly, we propose another variance loss to adjust the attention distribution globally. Ideally, the same locations should be assigned a relatively high attention weight once at most. Different from the coverage mechanism (See et al., 2017; Tu et al., 2016) tracking attention distributions of previous timesteps, we maintain the sum of attention distributions over all decoder timesteps, denoted as A. The i-th value of A represents the accumulated attention that the input state at i-th position has received throughout the whole decoding process. Without repeated high attention being paid to the same location, the difference between the sum of attention weight and maximum attention weight of i-th input state among all timesteps should be small. Moreover, the whole distribution of the difference over all input positions should have a flat shape. Similar to the definition of local variance loss, the global variance loss is formulated as:

$$g_i = \sum_t (a_{ti}^r) - \max_t(a_{ti}^r) \qquad (8)$$

$$\mathcal{L}_G = \frac{1}{|D|} \sum_{i=1}^{|D|} (g_i - \hat{g})^2 \qquad (9)$$

where g_i represents the difference between the accumulated attention weight and maximum attention weight at i-th position.

2.4 Model Training

The model is firstly pre-trained to minimize the maximum-likelihood loss, which is widely used in sequence generation tasks. We define $y^* = \{y_1^*, \cdots, y_T^*\}$ as the ground-truth output sequence for a given input sequence x, then the loss function is formulated as:

$$\mathcal{L}_{MLE} = -\frac{1}{T} \sum_{t=1}^{T} \log(p(y_t^*|x)) \qquad (10)$$

After converging, the model is further optimized with local variance loss and global variance loss. The mix of loss functions is:

$$\mathcal{L} = \mathcal{L}_{MLE} + \lambda_1 \mathcal{L}_L + \lambda_2 \mathcal{L}_G \qquad (11)$$

where λ_1 and λ_2 are hyper-parameters.

3 Experiments

3.1 Preliminaries

Dataset and Metrics. We conduct our model on the large-scale dataset CNN/Daily Mail (Hermann et al., 2015; Nallapati et al., 2016), which is widely used in the task of abstractive document summarization with multi-sentences summaries. We use the scripts provided by See et al. (2017) to obtain the non-anonymized version of the dataset without preprocessing to replace named entities. The dataset contains 287,226 training pairs, 13,368 validation pairs and 11,490 test pairs in total. We use the full-length ROUGE F1[1] and METEOR[2] as our main evaluation metrics.

Implementation Details. The data preprocessing is the same as PGN (See et al., 2017), and we randomly initialize the word embeddings. The hidden states of the encoder and the decoder are both 256-dimensional and the embedding size is also 256. Adagrad with learning rate 0.15 and an accumulator with initial value 0.1 are used to train the model. We conduct experiments on a single Tesla P100 GPU with a batch size of 64 and it takes about 50000 iterations for pre-training and 10000 iterations for fine-tuning. Beam search size is set to 4 and trigram avoidance (Paulus et al., 2018) is used to avoid trigram-level repetition. Tuned on validation set, λ_1 and λ_2 in the loss function (Equation. 11) is set to 0.3 and 0.1, respectively.

3.2 Automatic Evaluation Result

As shown in Table 1 (the performance of other models is collected from their papers), our model exceeds the PGN baseline by 3.85, 2.1 and 3.37 in terms of R-1, R-2 and R-L respectively and receives over 3.23 point boost on METEOR. FastAbs (Chen and Bansal, 2018) regards ROUGE scores as reward signals with reinforcement learning, which brings a great performance gain. DCA (Celikyilmaz et al., 2018) proposes deep communicating agents with reinforcement setting and achieves the best results on CNN/Daily Mail. Although our experimental results have not outperformed the state-of-the-art models, our model has a much simpler structure with fewer parameters. Besides, these simple methods do yield a boost

---

[1] We use the official package pyrouge https://pypi.org/project/pyrouge/
[2] http://www.cs.cmu.edu/~alavie/METEOR/

1224

Models	ROUGE-1	ROUGE-2	ROUGE-L	METEOR
PREVIOUS WORKS				
PGN (See et al., 2017)	36.44	15.66	33.41	16.65
PGN+Coverage (See et al., 2017)	39.53	17.28	36.38	18.72
Intra-att.+RL (Paulus et al., 2018)	39.87	15.82	36.90	-
FastAbs+RL (Chen and Bansal, 2018)	40.88	17.80	38.54	20.38
DCA+RL (Shi et al., 2018)	41.69	19.47	37.92	-
OUR MODELS				
PGN (ours)	36.72	15.76	33.40	17.19
PGN+Coverage (ours)	39.75	17.42	36.36	19.73
PGN+ARU	37.41	16.01	34.05	18.03
+Local variance loss	39.45	17.26	35.99	19.02
+Global variance loss	40.29	17.76	36.78	19.88

Table 1: Performance on CNN/Daily Mail test dataset.

in performance compared with PGN baseline and may be applied on other models with attention mechanism.

We further evaluate how these optimization approaches work. The results at the bottom of Table 1 verify the effectiveness of our proposed methods. The ARU module has achieved a gain of 0.97 ROUGE-1, 0.35 ROUGE-2, and 0.64 ROUGE-L points; the local variance loss boosts the model by 3.01 ROUGE-1, 1.6 ROUGE-2, and 2.58 ROUGE-L. As shown in Figure 2, the global variance loss helps with eliminating n-gram repetitions, which verifies its effectiveness.

3.3 Human Evaluation and Case Study

We also conduct human evaluation on the generated summaries. Similar to the previous work (Chen and Bansal, 2018; Nallapati et al., 2017), we randomly select 100 samples from the test set of CNN/Daily Mail dataset and ask 3 human testers to measure *relevance* and *readability* of each summary. Relevance is based on how much salient information does the summary contain, and readability is based on how fluent and grammatical the summary is. Given an article, different people may have different understandings of the main content of the article, the ideal situation is that more than one reference is paired with the articles. However, most of summarization datasets contain the pairs of article with a single reference summary due to the cost of annotating multi-references. Since we use the reference summaries as target sequences to train the model and assume that they are the gold standard, we give both articles and reference summaries to the annotator to

Models	Relevance	Readability
Reference	5.00	5.00
PGN	2.27	4.30
PGN+Coverage	2.46	4.88
Our model	2.74	4.92

Table 2: Human Evaluation: pairwise comparison between our final model and PGN model.

score the generated summaries. In other words, we compare the generated summaries against the reference ones and the original article to obtain the (relative) scores in Table 3. Each perspective is assessed with a score from 1 (worst) to 5 (best). The result in Table 2 demonstrate that our model performs better under both criteria w.r.t. See et al. (2017). Additionally, we show the example of summaries generated by our model and baseline model in Table 3. As can be seen from the table, PGN suffers from repetition and fails to obtain the salient information. Though with coverage mechanism solving saliency and repetition problem, it generates many trivial facts. With ARU, the model successfully concentrates on the salient information, however, it also suffers from serious repetition problem. Further optimized by the variance loss, our model can avoid repetition and generate summary with salient information. Besides, our generated summary contains fewer trivial facts compared to the PGN+Coverage model.

4 Related Work

The exploration on document summarization can be broadly divided into extractive and abstractive summarization. The extractive methods (Nalla-

Article: poundland has been been forced to pull decorative plastic easter eggs from their shelves over fears children may choke - because they look like cadbury mini eggs . **trading standards officials in buckinghamshire and surrey raised the alarm** over the chinese made decorations , as **they were ' likely to contravene food imitation safety rules '** . the eggs have now been withdrawn nationwide ahead of the easter break . scroll down for video . poundland has been been forced to pull decorative plastic easter eggs from their shelves over fears they may choke - because they look like cadbury mini eggs -lrb- pictured is the poundland version -rrb- . **the eggs bear a striking similarity to the sugar-coated chocolate treats** with a brown ' speckle ' designed to make it look like a quail 's egg -lrb- cadbury mini eggs are pictured -rrb- . . . . ' parents should also be wary of similar products being offered for sale over the easter period at other stores or online . '
Reference Summary: Trading standards officials in buckinghamshire and surrey raised alarm. Officers said they were 'likely to contravene food imitation safety rules'. The eggs bear a striking similarity to the sugar-coated chocolate treats.
PGN: Poundland has been forced to pull decorative plastic easter eggs from their shelves over fears children may choke - because they look like cadbury mini eggs. The eggs have now been withdrawn nationwide ahead of the easter break. The eggs have now been withdrawn nationwide ahead of the easter break.
PGN+Coverage: Trading standards officials in buckinghamshire and surrey raised the alarm over the chinese made decorations , as they were ' likely to contravene food imitation safety rules ' the eggs have now been withdrawn nationwide ahead of the easter break . the eggs bear a striking similarity to the sugar-coated chocolate treats with a brown ' speckle ' designed to make it look like a quail 's egg .
+ ARU: Eggs bear a striking similarity to the sugar-coated chocolate treats with a brown 'speckle' designed to make it look like a quail's egg. The eggs bear a striking similarity to the sugar-coated chocolate treats with a brown 'speckle' designed to make it look like a quail's egg.
+ Variance loss: Trading standards officials in buckinghamshire and surrey raised the alarm over the chinese made decorations, as they were 'likely to contravene food imitation safety rules'. The eggs have now been withdrawn nationwide ahead of the easter break. The eggs bear a striking similarity to the sugar-coated chocolate treats with a brown 'speckle'.

Table 3: The **bold** words in *article* are salient parts contained in *reference summary*. The blue words are salient information and the red words are repetition.
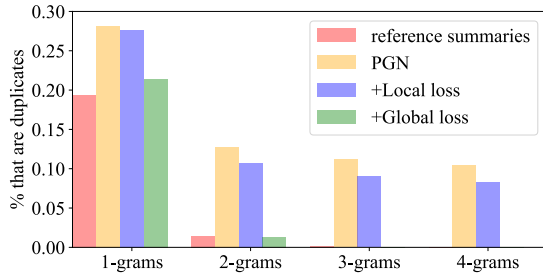

Figure 2: With global variance loss, our model (green bar) can avoid repetitions and achieve comparable percentage of duplicates with reference summaries.

pati et al., 2017; Jadhav and Rajan, 2018; Shi et al., 2018) select salient sentences from original document as a summary. In contrast, abstractive summarization (Rush et al., 2015; Nallapati et al., 2016; See et al., 2017; Chen and Bansal, 2018) generates summaries word-by-word after digesting the main content of the document. Out-of-vocabulary(OOV), repetition, and saliency are three conspicuous problems need to be well solved in abstractive document summarization. Some works (Nallapati et al., 2016; See et al., 2017; Paulus et al., 2018) handle the OOV problem by introducing the pointer network. See et al. (2017) introduces a coverage mechanism, which is a variant of the coverage vector (Tu et al., 2016) from Neural Machine Translation, to eliminate repetitions. However, there are just a few studies on saliency problem (Tan et al., 2017; Shi et al., 2018; Gehrmann et al., 2018). To obtain more salient in-

formation, Chen et al. (2016) proposes a new attention mechanism to distract them in the decoding step to better grasp the overall meaning of input documents. We optimize attention using an attention refinement unit under the novel variance loss supervision. As far as we know, we are the first to propose explicit losses to refine the attention model in abstractive document summarization tasks. Recently many models (Paulus et al., 2018; Celikyilmaz et al., 2018; Chen and Bansal, 2018; Zhou et al., 2018; Jiang and Bansal, 2018) have emerged taking advantage of reinforcement learning (RL) to solve the discrepancy issue in seq2seq model and have yielded the state-of-the-art performance.

5 Conclusion

In this paper, we propose simple but effective methods to optimize the vanilla attention mechanism in abstarctive document summarization. The results on CNN/Daily Mail dataset demonstrate the effectiveness of our methods. We argue that these simple methods are also adaptable to other summarization models with attention. Further exploration on this and combination with other approaches like RL remains as our future exploration. Besides, we will also conduct experiments on several other current summarization datasets like New York Times (NYT) (Paulus et al., 2018) and Newsroom (Grusky et al., 2018).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 2018. Deep communicating agents for abstractive summarization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1662–1675. Association for Computational Linguistics.

Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, and Hui Jiang. 2016. Distraction-based neural networks for modeling documents. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, pages 2754–2760. AAAI Press.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–686. Association for Computational Linguistics.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380. Association for Computational Linguistics.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109. Association for Computational Linguistics.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 708–719.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1693–1701.

T. Huang, G. Yang, and G. Tang. 1979. A fast two-dimensional median filtering algorithm. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(1):13–18.

Aishwarya Jadhav and Vaibhav Rajan. 2018. Extractive summarization with swap-net: Sentences and words from alternating pointer networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 142–151. Association for Computational Linguistics.

Yichen Jiang and Mohit Bansal. 2018. Closed-book training to improve summarization encoder memory. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4067–4077.

Yunjae Jung, Donghyeon Cho, Dahun Kim, Sanghyun Woo, and In So Kweon. 2018. Discriminative feature learning for unsupervised video summarization. *CoRR*, abs/1811.09791.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pages 3075–3081.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290. Association for Computational Linguistics.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083. Association for Computational Linguistics.

Shiv Shankar, Siddhant Garg, and Sunita Sarawagi. 2018. Surprisingly easy hard-attention for sequence to sequence learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 640–645.

Jiaxin Shi, Chen Liang, Lei Hou, Juanzi Li, Zhiyuan Liu, and Hanwang Zhang. 2018. Deepchannel: Salience estimation by contrastive learning for extractive document summarization. *CoRR*, abs/1811.02394.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 3104–3112, Cambridge, MA, USA. MIT Press.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1171–1181. Association for Computational Linguistics.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 2692–2700.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 2048–2057.

Qingyu Zhou, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou, and Tiejun Zhao. 2018. Neural document summarization by jointly learning to score and select sentences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–663. Association for Computational Linguistics.